基于 Petri 网的 APT 攻击模型生成方法 *

杜镇宇，刘方正，李翼宏

(国防科技大学 电子对抗学院，合肥 230037)

摘　要：在严峻的 APT（advanced persistent threat）攻击防御背景下，针对现有网络攻击建模方法无法反映 APT 攻击的攻击特点的问题，建立了基于 Petri 网的 APT 攻击模型。借助 Petri 网，首先针对 APT 攻击的特点及生命周期，建立 APT 攻击的基本 Petri 网模型；然后设计并实现针对具体 APT 攻击的 APTPN（advanced persistent threat petri nets）模型的生成算法，该算法能够生成具体 APT 攻击的完整的攻击路径，并能够对 APT 攻击进行检测及预测；最后实验通过模拟极光攻击验证了算法的有效性及正确性，并能够根据收集到的报警信息预测攻击者下一步的攻击手段。

关键词：Petri 网；APT；APTPN；建模；攻击路径

中图分类号：TP309.2　　doi: 10.3969/j.issn.1001-3695.2018.01.0041

Modeling method for advanced persistent threat based on Petri

Du Zhenyu, Liu Fangzheng, Li Yihong

(*Electronic confrontation Institute*, *National University of Defense Technology University*, *HeFei* 230037, *China*)

Abstract: Against the background of severe APT attack defense, aiming at the fact that the existing network attack modeling methods can not reflect the attack features of APT attacks, this paper established an APT attack model based on Petri nets. With Petri nets, it first established the basic Petri net model of APT attacks according to the characteristics and life cycle of APT attacks. Then, it designed and implemented the algorithm of generating APTPN (advanced persistent threat petri nets) model to generate its complete attack path against a specific APT attack. Finally, experiments verify the effectiveness and correctness of the algorithm by simulating auroral attacks.

Key words: Petri nets; advanced persistent threat; advanced persistent threat petri nets; modeling; attack path

0　引言

自 2010 年开始，APT（advanced persistent threat）攻击成为取代传统黑客攻击的一种重要的攻击手段，进入网络安全工作者的视野，并呈现出愈演愈烈的趋势[1]。

对于日益严肃的网络安全态势，作为受害者，应采取手段进行积极防御。现行的 APT 防御流程主要以被动防御为主[2~4]，即在危害发生之后采取手段对其进行检测防御。而当前 APT 威胁由于危害程度高、打击速度快等特性，被动防御的思想及手段已不能满足其防御态势。

因此，如何变被动防御为主动防御，在攻击任务达成之前根据攻击模型对攻击路径及发生的可能性进行威胁估计是当前应重点研究的问题，这其中的首要问题是建立一个合适的攻击模型。

Petri 网[5,6]作为分布式系统建模和分析的工具，具有严格数学定义和强大的图形表达能力。Petri 网不仅能够描述过程的静态结构，还可以模拟过程运行中动态行为，对于具有并发、异步等性质的信息系统，可以利用 Petri 网进行有效描述和分析。

同时，使用 Petri 网在对 APT 攻击建模上具有以下几点好处：a）通过对 Petri 网中库所及变迁节点的映射，可以对 APT 攻击手段、攻击流程进行详细的建模，经过映射后的 Petri 网对于攻击手段刻画的粒度更细；b）Petri 网可以添加时间序列，适应于 APT 攻击具有潜伏时间长、攻击阶段明显的特点；c）Petri 网可以添加颜色集，为 APT 攻击状态中的攻击序列的完成程度提供很好的描述手段，使建立的 APT 攻击模型能直观地反映出当前威胁级别，级别越高的威胁，可以作出优先响应，从而提高 APT 攻击防御能力；d）Petri 网可以对有向弧进行赋值，因而可以对建立好的 APT 攻击行为 Petri 网的每条攻击路径、转换关系赋值；e）使用 Petri 网对 APT 攻击流程、生命周期建模，能够克服以往建模方式中存在大量循环路径的问题。因此，本文选择 Petri 网建立 APT 攻击模型。

罗森林等人[7]以漏洞为基本研究对象，提出一种基于时间 Petri 网的渗透测试攻击模型构建方法，通过单漏洞利用模型整合成渗透测试攻击模型。黄光球等人[8]提出一种针对信任攻击

收稿日期：2018-01-24；**修回日期：**2018-03-09　**基金项目：**国家自然科学基金资助项目（U1636201）

作者简介：杜镇宇（1996-），女，黑龙江绥化人，硕士，主要研究方向为信息安全技术（dzyu1108@163.com）；刘方正（1983-），男，讲师，博士，主要研究方向为信息安全技术；李翼宏（1994-），男，硕士，主要研究方向为信息安全技术.

的面向对象 Petri 网建模技术，利用改写后的信任关系重建规则定义信任攻击 Petri 网，模拟了信任攻击的场景。吴迪等人[9]提出一种基于着色 Petri 网（CPN）的原子攻击的建模方法，基于主机权限及主机脆弱性节点定义一个基本的原子攻击的 Petri 网模型。上述基于 Petri 网的网络攻击建模方法完成了对单一网络攻击的建模，包括渗透测试攻击、信任攻击以及原子攻击，而这些建模方法无法针对 APT 攻击特点对完成对 APT 攻击的建模，其他的针对 APT 攻击的建模方法也存在一定问题。黄永洪等人[10]基于攻击图提出一种 APT 风险属性攻击图模型的构建方法，而该方法却没有给出合理的威胁表示。王辉等人[11]将贝叶斯理论引入网络攻击图，基于贝叶斯网络攻击图模型对网络攻击内部状态进行形象化描述，虽然给出了威胁表示，但是仅针对于简单的原子攻击，对于复杂的 APT 攻击无能为力。

因此，本文针对 APT 的阶段性特点及其生命周期，基于 Petri 网，建立 APT 攻击的 Petri 网模型，同时以该模型为基础，提出一种 APT 攻击路径生成算法，生成 APT 攻击路径。该算法能够自动生成 APT 攻击元路径序列，通过该序列能够直观地显示攻击者的攻击意图及攻击方式，以便提前防御，缩短 APT 防御周期，有效应对 APT 攻击。

1 基于 Petri 网的 APT 攻击模型

1.1 APT 攻击阶段性特点分析

杀链 IKC（intrusion kill chain）模型[11]使用攻击链来同时描述 APT 攻击的各个阶段以及其攻击手段，存在两点不足：a）其对于 APT 攻击阶段的划分存在缺漏，即缺少撤收阶段，通常，在 APT 攻击过程中，攻击者在完成攻击之后，会进行日志清除、文件损毁、注册表重写等一系列动作来防止被溯源，而这个过程中也会产生一定的可分析行为，并可作为判别依据加入到检测系统中去；b）该模型对于攻击阶段的考虑的粒度过细，而从攻击方式方面考虑的粒度过粗。

本文基于 IKC，改进其完整性不足以及存在冗余的问题，对 APT 的攻击阶段进行了重新划分，如图 1 所示。

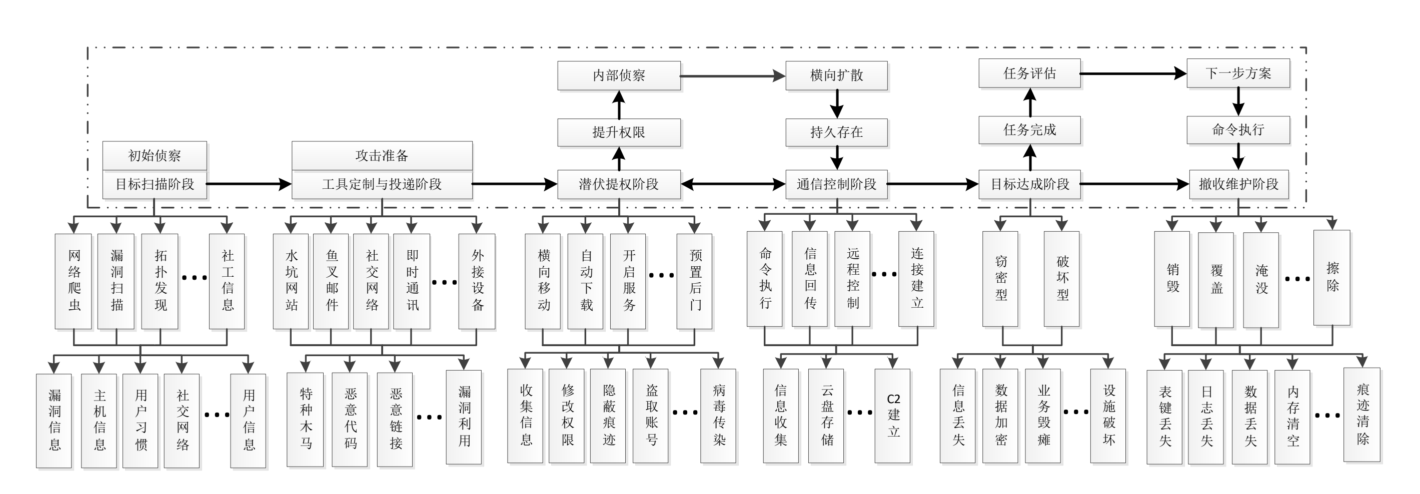


图 1 APT 攻击阶段划分模型

1）目标扫描阶段 通过网络爬虫、网络扫描、漏洞探测、拓扑发现、社会工程学信息等主要技术对被攻击目标的节点属性、漏洞信息、用户习惯、主机信息、弱口令、系统安装的防火墙、IDS、杀毒软件等相关信息进行探测，为下一步攻击提供信息支撑。

2)工具定制与投递阶段 通过对上一阶段扫描得到的相关信息的进一步分析整合，针对攻击目标的特点量身定做合适的攻击工具，具体包括特种木马、恶意代码等，并将这些隐藏到网页链接、PDF、Office、外接存储设备等。在这个阶段，攻击者会依靠上一阶段的用户习惯来选择攻击工具。

（a）当被攻击者经常访问某一特定网站，但其访问方式大多依赖于搜索引擎时，攻击者可制作另一相关网站，并将攻击工具夹带到该网站上，引诱受害者点击。通常该网页仅对被害群体可见，而其他人不在钓鱼名单上的不相关人员则显示为 404 error。

（b）当被攻击者经常使用邮件进行通信时，可以将攻击工具隐匿在邮件中，邮件内容通常是受害者感兴趣的话题，或者伪装成相熟人将该钓鱼邮件发送给被攻击者，引诱其点击附件中的链接。

（c）当被攻击者经常使用外界存储设备转移数据时，如 USB 存储设备、外接投影仪、打印机等，可将攻击工具事先加载到这些外接设备上，当受害者将该被感染的外接设备接入时，恶意代码等会自动释放到被攻击主机上，实现攻击工具的投放。

3）潜伏提权阶段 在该阶段中，攻击者通过上一阶段投放的恶意代码打开与用户通信的渠道，进一步操作提升攻击者权限为下一步做铺垫。通常在上一阶段夹带或隐匿的恶意代码由于受到用户操作等客观限制，其大小及功能都只能是微小的，其目的只是打开一个小小的缺口，让攻击者能够执行下一步操作。因此，在该阶段有三种常见情况：

（a）带有横向移动功能的恶意代码会在受害者不知情的情况下感染受害主机的其他磁盘、文件系统，搜寻敏感文件，等待回传。

（b）带有自动下载功能的恶意代码会在用户联网操作时打开事先写好的恶意链接，下载并安装具有更强功能的恶意代码，

可能是记录键盘操作、拷贝日志信息、修改注册表项、扫描网段内其他主机并收集信息等一切有利于攻击者下一步操作的功能。

（c）带有开启服务功能的恶意代码会打开受害主机的远程控制服务、文件传输端口、数据库操作端口等相关服务，盗取用户账号信息等实现对目标主机的远程控制。

在这一阶段里，若直接攻击的目标主机不能完成攻击目的，可以实现目标的扩展移动，通过对上述操作的全部循环，或以该受害主机做跳板机通过对上述操作的部分循环入侵内网其他主机。

4）通信控制阶段　攻击者通过上述阶段的积累已经实现对目标主机的控制，因此在该阶段，攻击者开始实现与受害主机的通信控制，可以远程执行命令，对收集的数据进行筛选并选择与攻击目标相符的相关数据。

5）目标达成阶段　在这一阶段攻击者完成了对攻击目标的所有攻击操作，开始实现攻击目标，达成任务目的。针对任务目的不同，主要有两个方面：

（a）窃密性 APT 攻击。这类攻击主要以窃取敏感数据为目的，因此该阶段完成对敏感文件的回传操作。

（b）破坏型 APT 攻击。这类攻击主要以影响、干扰、破坏目标的正常运行为目的，通过删除文件、修改用户权限、破坏内网主机等毁坏操作达成任务目的。

6）撤收维护阶段　APT 攻击是隐蔽的，有的受害主机遭受攻击而不知情，当后果严重时才会有所反映。通常攻击者为防止被溯源或者被阻止，会为自己留有撤退通道，方便攻击者撤回。这个阶段，攻击会销毁攻击工具、擦除入侵痕迹、覆盖攻击数据、删除日志记录、淹没内存等来完成攻击的撤收。其次，若攻击目标是一个长期目标，攻击者在完成阶段性目标后，会掩盖侵入痕迹、潜伏在目标主机中，继续保持对目标的监控，等待下一次攻击的开始。

1.2　模型建立

基于 Petri 网的 APT 攻击模型是根据 APT 攻击的阶段性特点，针对 APT 攻击过程中的攻击流程、使用的攻击方法及攻击方式建立的。通过将 APT 攻击中的基本要素与 Petri 网中基本元素的映射，能够建立出包含 APT 攻击路径的 APT 攻击模型。映射关系见表 1。

表 1　基本元素映射关系

Petri 网	APT 攻击
库所	攻击者资源以及攻击目标状态
变迁	攻击手段
有向弧	攻击状态转换

APT 攻击的 Petri 网模型是一个九元组 APTPN（advanced persistent threat petri nets），公式形式记为

APTPN=<P，T，F，W，L，G，t，H，K>

其中：

P={Pij|i=0,1,2,⋯,6;j=0,1,2,⋯,N}，库所集。一个库所包含两个部分，一是攻击目标当前状态，二是攻击者获得资源。表示为二元组的形式，即 Pij=<Sij,Vij>，且 Sij={aj|j=1,2,⋯,n}、Vij={ai|i=1,2,⋯,n}，aj 是主机与初始状态相比，发生改变的资源编号；ai 是攻击发起方获得的资源编号。当攻击目标是一台主机时，其攻击目标状态即为该主机状态。同时，i 为所处阶段编号，j 为所处子阶段编号。

库所分为主库所及辅助库所两类。通常 APT 攻击不是单个攻击源的动作，而是由多个攻击源共同完成的合作攻击，辅助库所用来表示由其他攻击源完成的对其主库所之后的变迁的补充。

同时，针对 APT 攻击的阶段性特点，将模型根据上述阶段进行划分，在每个阶段或子阶段的开始及结束均有一个库所节点，用于标志其开始状态及结束状态。APT 攻击模型中的库所含义见表 2。

表 2　库所含义及对应关系

库所 P		库所含义	
		S（攻击目标当前状态）	V（攻击者获得资源）
P0		S0，攻击受害方的初始状态。	V0=Ø
	P11	无变化，S0	V11={a1(生活习惯),a2(常用社交网络),a3(电话号码),a4(邮箱账号)...}
	P12	无变化，S12=S11=S0	V12={V11,a1(存活主机),a2(开放端口),a3(可利用漏洞)...}
...	...	...	...
P1		无变化，S1=⋯=S0	V1={V11,V12,⋯}
	P21	无变化，S21=S1=⋯=S0	V21={V1,a1(制作好的包含恶意程序的工具),⋯}
...	⋯	...	⋯
P2		有变化，S2={a1=(下载并安装恶意工具)}	V2={V21,⋯}
	P31	有变化，S31={S2,a1=(下载并安装恶意程序),a2=(脆弱节点暴露)...}	V31={V2,a1={攻击入口},a2={脆弱节点}⋯}
	P32	有变化，S32={S31,a1=(启动项被修改),a2=(管理员密码被修改),a3=(注册表被修改),a4=(被设置后门)...}	V32={V31,a1=(后门),a2=(当前管理员密码),a3=(远程服务端口号)⋯}
...	⋯	...	⋯
P3		无变化，S3={S32,⋯}	V3={V32,⋯}
	P41	有变化，S3={S3,a1(c1 服务器),⋯}	V41={V3,a1(c2 服务器),a2(主机中文档信息),a3(主机系统关键信息)⋯}
⋯	...	⋯	...
P4		无变化，S4={S41⋯}	V4={V41,⋯}
	P51	有变化，S51={S4,a1={信息丢失},a2={信息加密},a3={权限更改},⋯}	V51={V4,a1={重要信息},a2={主机权限},⋯}
⋯	...	⋯	...
P5		无变化，S5={S51⋯}	V5={V51,⋯}
	P61	有变化，S61=S0	V61={V5}
⋯	...	⋯	...
P6		无变化，S6={S61⋯}	V6={V61}

T={Tijk|i=0,1,2,…,6;j=1,2,…,M;k=1,2,…,M’}，变迁集。变迁是对当前 APT 攻击阶段中攻击方法的描述。变迁采用三位编号的形式，第一位为所处阶段编号、第二位为所处子阶段编号、第三位为所处变迁位置编号。具体见表 3。

表 3　变迁含义及对应关系

攻击阶段	攻击意图及主要行为		变迁编号及对应攻击方法			
	主要行为	攻击意图	变迁编号	攻击方法	变迁编号	攻击方法
目标扫描阶段 1	收集攻击目标的社会工程学信息	搜索、欺骗等	T111 T113 …	人肉搜索 网络钓鱼 …	T112 T114	社交网络 反向社工
目标扫描阶段 2	进一步收集目标网络的配置信息、拓扑结构等，以及系统用户的个人资料	扫描、嗅探等	T121 T123 T125	端口扫描 SQL 语句探测 网络爬虫	T122 T124 …	代码分析 Ping 存活主机 …
工具定制与投递	针对搜集目标信息制定攻击计划、定制攻击工具、定向投递	水坑、钓鱼等	T211 T213 T215	用户习惯 水坑网页 外接木马	T212 T214 …	钓鱼网站 恶意邮件 …
潜伏提权阶段 1	发现目标的脆弱节点，侵入目标系统，传播恶意程序	下载程序、打开端口等	T311 T313	横向移动 打开远程服务	T312 …	下载恶意程序 …
潜伏提权阶段 2	深入入侵目标节点，稳定侵入状态，帮助攻击者下一步攻击	搭建后门、修改系统状态等	T321 T323 T325	修改密码 变更角色组 预置后门	T322 T324 …	修改启动项 修改注册表 …
通信控制阶段	进入到攻击目标，查询所需信息，与攻击者远程通信	命令执行，数据回传	T411 T413	回传数据 远程执行指令	T412 …	流量异常 …
目标达成阶段	完成任务	窃密、破坏	T511 T513 T515	回传数据 删除文件 破坏内网主机	T512 T514 …	加密数据 修改权限 …
撤收维护阶段	防止被溯源、方便下一次攻击	清除痕迹	T611 T613 T615	销毁攻击工具 覆盖攻击数据 淹没内存	T612 T614 …	擦除入侵痕迹 删除日志记录 …

F={Fi|i=count(PxT，TxP)}，有向弧集。PxT 表示 P 是 T 的前驱库所；TxP 表示 P 是 T 的后继库所，同时集合中元素个数小于 M*N。

W:F→Q，权重函数。其中 Q 为有理数集，即 W 是有向弧 F 对有理数集的映射函数，取值表示有向弧的权重，是一个权值函数。表示为 W=[Wij]NxM，其中 N=|P|，M=|T|，则

$$\mathrm{Wij}=\begin{cases} w(T_i,P_j) & P_j\text{是}T_i\text{的后继库所} \\ -w(T_i,P_j) & P_j\text{是}T_i\text{的前驱库所} \\ 0 & \text{其他} \end{cases}$$

其中：$w(T_i,P_j)$、$w(P_i,T_j)$ 为弧标注值。

对于辅助库所来说，辅助库所的 W 矩阵表示方法是在 W 矩阵中添加列向量 Pr，表示一个变迁是否有辅助库所。若有则为 1；没有则为 0。对于为 1 的矩阵元素，建立其辅助库所关联矩阵 W’，其中 W’ 的定义方法与 W 一致。

L={AND，OR}，库所间逻辑关系集。能够描述当某一变迁的完成需要辅助库所时的库所间关系。且取值为 AND 时，表示当前两库所相互依赖，只有同时满足触发条件后，才可进行下一步操作。

G:G→PxT，取值为{0,1}，变迁触发条件。对于每一个变迁，该值由其前驱库所的状态决定，当前驱库所的状态满足触发条件，则其取值为 1，表示该变迁可以完成；否则为 0，继续等待前驱库所的状态变化。若其前驱库所不唯一，存在辅助库所且 L=AND 时，需当主库所及辅助库所的状态同时满足触发条件时，取值为 1。

t={tijk|i=0,1,2,…,6;j=1,2,…,M;k=1,2,…,M’}，变迁持续时间。描述从开始到结束所经历的时间，开始时刻定义为其前驱库满足变迁函数的时刻，结束时刻为其后继库所发生状态转移的时刻。

H={Hij|i=0,1,2,…,6;j=1,2,…,M;k=1,2,…,M’}，为变迁命题集。用来对变迁的状态进行判定，H→{0,1}。

K={Kij|i=0,1,2,…,6;j=0,1,2,…,N}，库所命题集。用来对库所的状态进行判定，K→{0,1}。

APT 攻击的 Petri 网模型的图形式见图 2。

1.3　模型分析

对于 Petri 网的分析，主要分析其性质，包括有界性、安全性、活性、守恒性、可达性以及覆盖性[12]。分析方法包括可达

树及可达图。周建涛等人[14]证明可达图相较于可达树在 Petri 网的活性及可达性分析上具有优势。图 3 是 APT 攻击的 Petri 网模型的可达图。

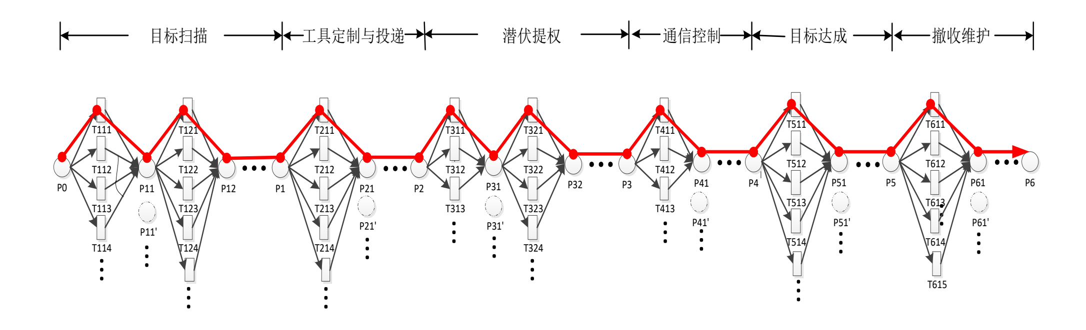


图 2　APT 攻击的 Petri 网模型



图 3　APT 攻击的 Petri 网模型的可达图

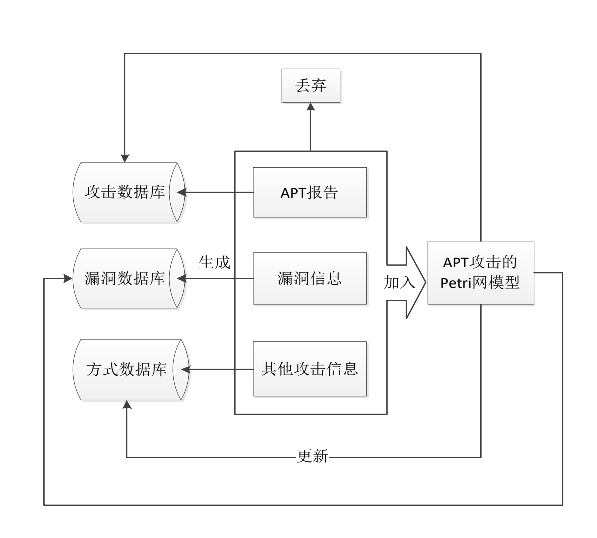


图 4　模型构建框架

APT 攻击的 Petri 网全模型是一个通用的模型，针对不同的 APT 攻击来说，其具体模型存在差异。对于一次 APT 攻击来说，其从攻击开始到攻击完成是在有限的攻击步骤内完成的，虽然对于某一个具体 APT 攻击的攻击步骤数量不同，但模型在库所和变迁的数量上均有界，不是无限的，因此，APT 攻击的 Petri 网模型可以处理模型中组合爆炸的问题。

在模型中，库所状态的转换是通过变迁触发完成的，其托肯值表示当前库所状态是否满足可以触发变迁的状态，其取值只有 0 和 1 两种，因而符合安全性的判定条件，所以 APT 攻击的 Petri 网模型满足安全性及有界性。同时，在该模型中，变迁是由攻击方式决定的，只有攻击发生，其变迁存在，由变迁发生变换的库所存在，因而模型不存在不可达的库所节点，模型具有可达性。

模型构建的过程是一个动态过程。现有的建模技术，其生成的攻击模型具有自限性，当出现变种攻击或新的攻击行为时，模型中缺乏应对方法。而基于 Petri 网生成的 APT 攻击模型是对其现有 APT 攻击中使用的攻击手段等进行整理，整合现有 APT 攻击特点建立的，因而模型的构建应能够动态调整，当出现新的 APT 攻击行为时，模型能够自适应地进行加入、融合或丢弃。图 4 是模型构建框架。

构建过程是：首先通过现存的 APT 报告、漏洞信息以及其他攻击信息生成基础的攻击数据库、漏洞数据库以及方式数据库；然后根据基础的三大数据库构建 APT 攻击模型；最后当有新的信息被发现并被收集时，依据类别判断其是否在基础数据库中。如果在，则丢弃该条信息；若不在，则将该信息加入到模型中，同时更新数据库。

攻击数据库是包含 APT 报告在内的能够反映一次 APT 攻击完整信息的数据库。在该数据库中包含是 APT 组织名称、APT 报告附件、提取的 APT 攻击特征三个部分。

漏洞数据库则依托于 CVE（common vulnerabilities and exposures）等漏洞数据库建立，但其内容上只包含 APT 攻击中利用的漏洞。这可能存在两种情况：a)漏洞已知，对于已知的漏洞利用，则根据从 CVE 中查询到的信息输入到漏洞数据库中；b)漏洞未知，若 APT 攻击使用的是 0day 漏洞或尚未披露的漏洞，则根据截获到的样本以及报告等将漏洞名称、利用方式、危害程度输入到漏洞数据库中。

方式数据库中的主要内容是 APT 攻击中的 TTPs（tactics、techniques & procedures）特征，是攻击者的攻击方式及攻击手法的表示。以 APT 攻击中的阶段性特点对 TTPs 特征进行区分，方式数据库中包含攻击者各个阶段表现出的 TTPs 特征。

在模型应用方面，该模型能够有利于对于 APT 攻击的检测和预测。在检测上，图 5 所示的元模型的应用中，通过引入信息集{$Alert_1$,$Alert_2$,…$Alert_{i+X}$}，可利用模糊判别法[14]或构建逻辑表达式法[15]或专家系统来判定模型中某一库所 P_X 是否已发生或发生的概率 $P(P_X)$，进而通过已判定完成的多个库所来判定包含在库所在内的 APT 攻击是否发生。在预测上，根据判定的已发生库所，能够在模型中找到该库所的一个或多个后继变迁 T_{x+i}，通过识别这些变迁所表示的攻击手段，能够为安全工作者提供预判，以便于提前部署，主动防御。

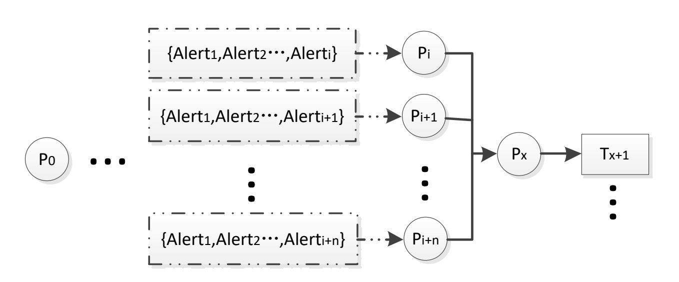


图 5 APT 攻击的 Petri 网元模型的应用

2 APTPN 模型生成算法

一条 APT 攻击路径是指由攻击元路径组成的序列，记为：{ (P0,T1,P1), (P1,T2,P2), … ,(Pi,Tj,Pi+1), (Pi+1,Tj+1,Pi+2) … ,(Pi+d,Tj+d,Pi+d+1)}。其中(Pi,Tj,Pk)为攻击元路径，Pi 与 Tj 连通，Tj 与 Pi+1 连通。

当 Pk 为撤收完成节点时，此时攻击路径为一条完整的攻击路径。即从攻击起点—目标选定开始到攻击终点—撤收完成结束的一系列攻击元路径组成的序列。

例如图 3 中的红色路径即表示一条完整的 APT 攻击路径，可表示为：{ (P0,T111,P11), (P11,T121,P1) ,(P1,T211,P2), (P2,T311,P31), (P31,T321,P3), (P3,T411,P4), (P4,T511,P5), (P5,T611,P61)}。

2.1 算法基本思想

算法的目的是给定库所和变迁，生成库所与变迁之间的有向弧关系，进而生成一条完整的 APT 攻击路径，最后建立完整的针对某一具体 APT 攻击的 APTPN 模型。算法主要分成三个部分：一是信息预处理，生成连通序列集、变迁序列集以及库所序列集；二是逻辑推理攻击元序列集；三是攻击路径生成。算法的基本思路图如图 6 所示。

连通序列由多个连通元序列组成。连通序列可描述目标网络中主机间连通情况，记为{<SIP1,DIP1,1>,<SIP2,DIP2,1>, … ,<SIPn,DIPn,1>}。其中：<SIPi,DIPi,1>为连通元序列。

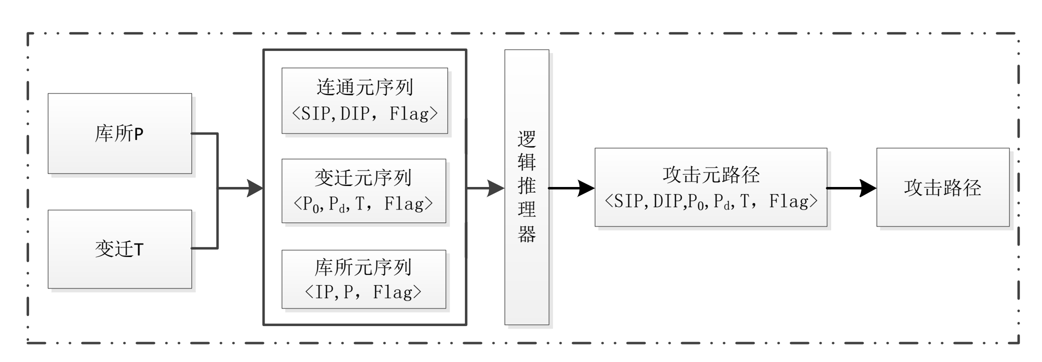


图 6 基于 Petri 网模型的 APT 攻击路径生成算法基本思想

连通元序列是指，当源主机可向目的主机发送数据包时，称源主机与目的主机连通。记为<SIP,DIP，Flag>。其中：SIP 为源 IP。

DIP 为目的 IP。

Flag 为标志位，当为 1 时，表示该元序列为连通元序列。

通过该连通序列可以很好地描述主机之间的连通情况。该序列是基于网络拓扑图、存活主机信息、防火墙配置等相关信息建立的，通过对上述信息的整合生成能够描述现行网络环境的连通性。

变迁序列由多个变迁元序列组成。变迁序列可描述变迁发生前后库所状态变化，记为{<IP1,P01,Pd1,T1,2>，<IP2,P02,Pd2,T2,2>，…，<IPn,P0n,Pdn,Tn,2>}。其中<IPi,P0i,Pdi,Ti,2>为变迁元序列。

变迁元序列是指，当变迁发生的主机库所状态达到 P0 时，变迁 T 可发生，且发生后，库所状态为 Pd，记为<P0,Pd,T,Flag>。其中：

P0 为变迁发生的起点库所。

Pd 为变迁结束时的终点库所。

T 为变迁名称。

Flag 为标志位，当为 2 时，表示该元序列为变迁元序列。

通过该序列能够描述变迁发生的前提条件和结果状态。该序列的生成是基于漏洞数据库和变迁数据库实现的，其中漏洞数据库主要描述主机存在漏洞的相关信息，而变迁数据库是基于 APT 攻击的 Petri 网模型的变迁构建的，通过对这两个数据库的信息提取以及主机脆弱性和可能采取的攻击方式的匹配即能得到描述一次变迁的变迁序列。

库所序列由多个库所元序列组成。库所序列可描述目标网络中各个主机当前库所状态，记为{<IP1,P1，3>，<IP2,P2，3>，…,<IPn,Pn,3>}。其中<IPi,Pi，3>为库所元序列。库所元序列是指，主机 IP 当前的库所状态为 P，记为<IP,P，Flag>。其

中：IP 为主机 IP。

P 为该 IP 当前库所。

Flag 为标志位，当为 3 时，表示该元序列为库所元序列。

攻击元路径是通过推理得到的，表示当源 IP 的库所状态为 P0 时，变迁 T 发生，且导致目的 IP 的库所状态转换成 Pd，并称源 IP 与目的 IP 通过变迁 T 可达，记为<SIP,DIP,P0,Pd,T,Flag>。其中：

SIP 为起点 IP。

DIP 为终点 IP。

P0 为起点 IP 的库所。

Pd 为终点 IP 的库所。

T 为变迁名称。

Flag 为标志位，当为 4 时，表示为攻击元路径。

在逻辑推理器中共存在七种推理。对于路径中出现的前驱库所、变迁以及后继库所，若相同记为 0，不同记为 1。根据排列组合规律，共有 23=8 种组合方式，分别为 000、001、010、011、100、101、110、111。而 000 与 111 表示同一种情况，因此共有七种逻辑关系。图 7 是七种库所及变迁因果关系，分别是：

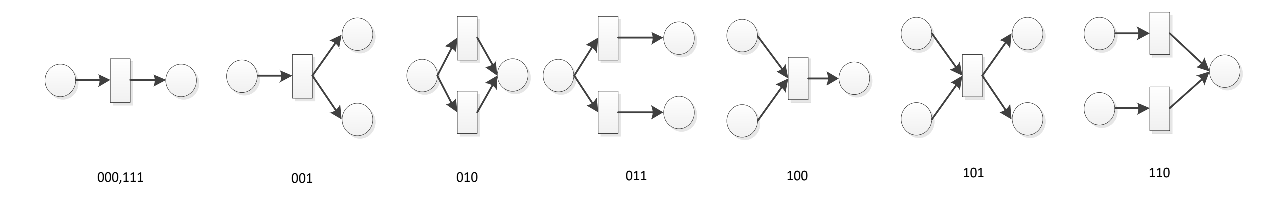


图 7　基于 Petri 网模型的 APT 攻击路径生成算法基本思想

同时，观察上述七种推理关系，可得到四种库所及变迁关系类型（图 8），即：

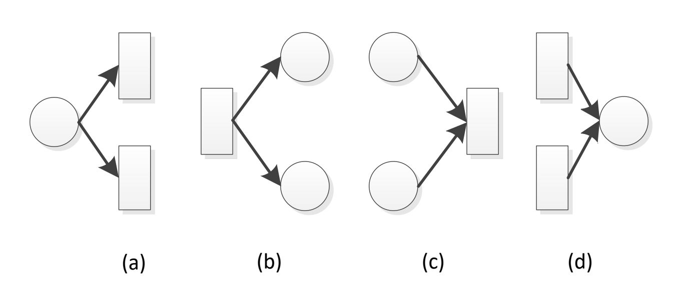


图 8　L=AND 的四种库所与变迁节点的关系

（a）一个库所可有多个后继变迁。一个系统状态节点后可能有多种攻击手段节点，通常在 APT 攻击中，某一系统状态下攻击者为达到下一攻击目标可使用的攻击方式不只一种，即在一个库所下可能有多个变迁。当其中一个变迁完成时，系统状态发生相应改变到一个库所节点，又会有多个变迁节点在其之后。

（b）一个变迁可有多个后继库所。一个攻击手段完成之后，可能有多个系统状态同时发生改变。例如，在暴力破解密码成功后，由于设密码时产生的惯性思维，受害者可能对不同主机或系统设置相同的密码，所以其破解得到的密码可不只匹配到一台主机或系统，即一个变迁发生后，可有多个库所状态发生改变。

（c）一个变迁可有多个前驱库所。一个攻击手段的完成可能不止需要一个系统状态，在 APT 攻击中，一个攻击任务的完成往往不是由单个人或组织完成的，而是由多个攻击源之间共享资源，分工合作完成的，因而一个变迁可能对应的前驱库所不止一个。

（d）一个库所可有多个前驱变迁。一个库所节点的状态变化可能需要由多个变迁同时发生才能达到触发条件。

在逻辑推理器中存在的其中推理如下：

推理 1：001	推理 2：010	推理 3：011	推理 7：000,111
前提：	前提：	前提：	前提：
<IP1,IP2,1>,<IP1,IP3,1>	<IP1,IP2,1>	<IP1,IP2,1>,<IP1,IP3,1>	<IP1,IP2,1>
<P1,P2,T1,2>,<P1,P3,T1,2>	<P1,P2,T1,2>,<P1,P2,T2,2>	<P1,P2,T1,2>,<P1,P3,T2,2>	<P1,P2,T1,2>
<IP1,P1,3>	<IP1,P1,3>	<IP1,P1,3>	<IP1,P1,3>
结果：	结果：	结果：	结果：
<IP1,IP2,P1,P2,T1,4>	<IP1,IP2,P1,P2,T1,4>	<IP1,IP2,P1,P2,T1,4>	<IP1,IP2,P1,P2,T1,4>
<IP1,IP3,P1,P3,T1,4>	<IP1,IP2,P1,P2,T2,4>	<IP1,IP3,P1,P3,T2,4>	Null
L=AND	L=AND	L=AND	L=OR
推理 4：100	推理 5：101	推理 6：110	
前提：	前提：	前提：	
<IP1,IP2,1>,<IP3,IP2,1>	<IP1,IP2,1>,<IP3,IP4,1>	<IP1,IP2,1>,<IP3,IP2,1>	
<P1,P2,T1,2>,<P3,P3,T1,2>	<P1,P2,T1,2>,<P3,P4,T1,2>	<P1,P2,T1,2>,<P3,P2,T2,2>	
<IP1,P1,3>,<IP3,P3,3>	<IP1,P1,3>,<IP3,P3,3>	<IP1,P1,3>,<IP3,P3,3>	
结果：	结果：	结果：	
<IP1,IP2,P1,P2,T1,4>	<IP1,IP2,P1,P2,T1,4>	<IP1,IP2,P1,P2,T1,4>	
<IP3,IP2,P3,P2,T1,4>	<IP3,IP4,P3,P4,T1,4>	<IP3,IP2,P3,P2,T2,4>	
L=AND	L=AND	L=AND	

同时，APT 攻击分为窃密型攻击及破坏型攻击两种。以窃密型攻击为例，若攻击者的攻击目标是位于目标主机上的秘密信息{M}，则初始时的库所状态应为<S,Ø>，而攻击任务达成时的库所状态应为<S-{M},{M}>。

最后，通过对攻击元路径集的深度优先遍历，即可得到完整的 APT 攻击路径。

2.2　算法设计与实现

算法伪代码如下：

//信息获取函数，输入是（连通信息表，漏洞信息表，端口信息表，地址信息表）

INS(thrs,vuls,pors,ips)

{

第一步：从地址信息表中未每一个地址以 ip 作为表头，新建一个空链表；

第二步：将表头的指针域指向输入信息中，以该 ip 为起点可达的终点 ip;将该连通信息从连通信息表中删去；计数器 1 加 1；

If 连通信息表中关于该 ip 的信息为空

更改表头 ip，返回第二步；

Else if 连通信息表为空

转到第三步；

Else

返回第二步

第三步：将当前指针域指向表头 ip 的漏洞信息；并将该条信息从漏洞信息表中删去；计数器 2 加 1；

If 漏洞信息表中关于该 ip 的信息为空

更改表头 ip，返回第三步；

Else if 连通信息表为空

转到第四步；

Else

返回第三步

第四步：将当前指针域指向表头 ip 的端口信息；并将该条信息从端口信息表中删去；计数器 3 加 1；

If 端口信息表中关于该 ip 的信息为空

更改表头 ip，返回第四步；

Else if 端口信息表为空

返回链表计数器 1,2,3 的值；

Else

返回第四步

}

//序列生成函数，输入是信息获取阶段生成的链表

SEQ(L)

{

第一步：取出链表中从表头开始的第一个指针到计数器 1 个指针所指的地址与表头生成连通序列；

第二步：取出链表中从计数器加 1 到计数器加计数器 2 个指针所指的漏洞编号与表头生成变迁序列；从变迁数据库中补充其他变迁序列；

第三步：构建库所序列矩阵；

}

//路径生成函数

SFS()

{

第一步：判断当前库所序列、变迁序列以及连通序列中的元素的推理情况编号；

第二步：依据当前推理编号推理出攻击元序列；

第三步：深度优先遍历攻击元序列集，得到攻击路径；

}

2.3　算法分析

算法主要包括三个函数，即信息获取函数、序列生成函数、路径生成函数。

信息获取包括连通信息、漏洞信息、端口信息以及地址信息。在存储形式上，算法采用链表的形式进行存储，即每一个主机及服务器（以下简称脆弱节点）单独形成一个链表。表头是脆弱节点的地址信息，后面则依次链接该脆弱节点可以访问的其他节点地址、该脆弱节点自身的漏洞编号、该脆弱节点自身开放的端口号。

序列生成包括连通序列、变迁序列及库所序列的生成。对于连通序列，从表头开始后的第二个单元取地址与表头形成连通序列。对于变迁序列，链表中存储的只包含漏洞信息，而实际的变迁还应包括目标扫描等阶段的信息，这部分的信息从变迁数据库中获取。库所序列与连通序列和变迁不同，库所序列是随着变迁的发生而逐渐变化的，因而构建库所序列矩阵。库所序列矩阵的行向量为脆弱节点，列向量为单一变迁，矩阵中的单个元素表示该脆弱节点在当前变迁发生之后的后继库所状态。

路径生成是算法的核心部分，该部分主要依据上文中的七个推理，对库所及变迁可能存在的七种关系生成攻击元路径。其核心是应用 IF 语句的判断。

算法的时间复杂度主要在序列集遍历上，采用深度优先遍历，若采用邻接表作为存储形式，则其时间复杂度为(|V|+|E|)；若采用邻接矩阵作为存储形式，其时间复杂度为ɵ（$|V|^2$）。而对于 APTPN 模型来说，遍历的是攻击元路径序列，其中模型的顶点数量即是库所序列中库所的数量 6N，边的数量即是变迁序列中变迁的数量 6MM’。因此，算法的时间复杂度可以控制在ɵ（N^2）以内，即 O（N^2）。

3　模拟实验及结果分析

3.1　实验设计

模拟实验的目的是检验 APTPN 模型生成算法的有效性及正确性。实验采取模拟已知的 APT 攻击——谷歌极光攻击来搭建实验局域网。通过使用模型生成算法生成该实验局域网的

APTPN 模型，验证算法的有效性。并通过将生成的 APTPN 模型与真实的极光攻击作对比，验证算法的正确性。

由于 APT 攻击具有针对性、持续性，当选定一个攻击目标后，攻击者会采取各种手段，不计成本地对其展开攻击。因此，实验的攻击目标与极光攻击的攻击目标——邮件服务器一致。

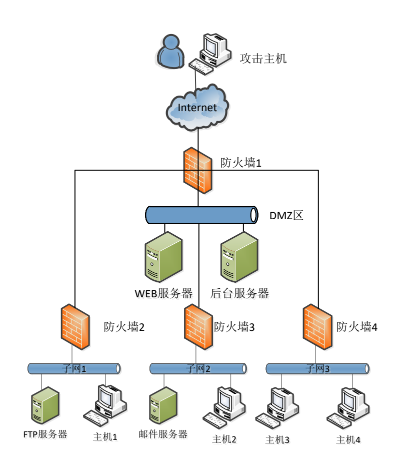


图 9　模拟极光攻击搭建局域网

实验模拟谷歌极光攻击，搭建如图 9 所示的局域网，各主机均建立在虚拟机上运行。为更加接近真实的网络情况，采用虚拟机桥接模式模拟 DMZ 区中主机，采用虚拟机 NAT 模式模拟子网 1，采用虚拟机 Host-only 模式模拟子网 2 及子网 3。本实验的攻击目标是子网 2 中的邮件服务器。具体的实验配置见表 4。

表 4　实验中使用主机及服务器配置情况

主机名称	网络模式	主机 IP	操作系统	编号
WEB 服务器	直连主机	192.168.42.88（DHCP）	Apache 2.4.2	H1
后台服务器	桥接模式	192.168.42.111（DHCP）	Win Server 2003 SP2	H2
FTP 服务器	NAT 模式	192.168.193.129（DHCP）	Win 2000 HFS Server	H3
主机 1	NAT 模式	192.168.193.130（DHCP）	Win XP SP3	H4
邮件服务器	Host-only 模式	192.168.117.222（手工）	Win XP SP3	H5
主机 2	Host-only 模式	192.168.117.111（手工）	Win XP SP3	H6
主机 3	Host-only 模式	192.168.120.111（手工）	Win 7 SP2	H7
主机 4	Host-only 模式	192.168.120.222（手工）	Win XP SP3	H8

由于实验中虚拟机的 NAT 模式只能设置一次，所以其他子网只能采用 Host-only 模式。而同时 Host-only 模式无法实现各子网间的连通，进而无法更加准确地模拟真实的网络环境。为解决上述问题，实验在子网 1 与 2 之间以及子网 2 与 3 之间设置网关，以实现子网间的连通。

以子网 2 与 3 之间的网关为例。实验中网关设置包括两个部分，一是为网关添加两个网卡，并设置 IP 及网关地址；二是打开路由转发服务并设置路由转发规则。

3.2　实验内容

首先获取实验局域网中的连通情况。局域网中的主机及服务器连通情况如下：局域网中 Web 服务器可以访问子网 1 中的主机 1；网 1 中的 FTP 服务器可以访问子网 2 及 3 中的主机及服务器，同时可以访问子网 1 中的主机 1；网 2 中的主机 2 可以访问邮件服务器及主机 3、4；子网 3 中的主机 3 可以访问主机 4；主机 1、FTP 服务器、主机 2 及 3 可以访问 Web 服务器和后台服务器。

然后获取实验局域网中的漏洞扫描结果。实验环境中能够探测到的漏洞多于表 5 所示的漏洞，表 5 中所示漏洞仅包含实验所利用的漏洞。其中条件列仅写出最低权限，而结果列仅写出最高权限。例如，若条件是外部用户，则 User 及 Root 权限时，条件也满足；若结果是 root，则结果也可达外部用户及 User 权限。其中权限关系为递进关系，即外部用户<User<Root。根据模型中库所定义以及局域网中主机及服务器的漏洞情况，获取实验中所包含的库所，见表 6。

根据局域网中的漏洞列表、APT 攻击的流程特点以及模型中的变迁定义，获取实验中的变迁及其前驱和后继库所，见表 7。

表 5　实验中各主机及服务器的漏洞列表

编号	CVE 编号	漏洞信息	条件	结果
H1	CVE-2013-2249	mod_session_dbd 模块安全漏洞	外部用户	Root
	CVE-2012-2687	HTML 注入漏洞	User	Root
	CVE-2011-3607	函数的整数溢出漏洞	User	Root
H2	CVE-2008-4250	服务器服务漏洞	外部用户	User
H3	CVE-2008-0407	认证漏洞	外部用户	User
	CVE-2014-0407	虚拟机组件漏洞	User	Root
H4	CVE-2013-3940	图形设备接口整数溢出漏洞	外部用户	User
	CVE-2013-5065	内核中执行并特权提升	User	Root
H5	CVE-2013-3346	执行 Reader 进程中的恶意代码	外部用户	User
	CVE-2014-4971	内存写入数据	外部用户	Root
H6	CVE-2013-3196	内核特权提升漏洞	User	Root
	CVE-2010-0249	极光攻击漏洞	外部用户	Root
H7	CVE-2010-1117	Explorer 8 堆缓冲区溢出漏洞	外部用户	User
	CVE-2015-0084	安全功能绕过漏洞	User	Root
H8	CVE-2012-1853	远程管理协议栈溢出漏洞	外部用户	Root

表 6　库所含义列表

	目标状态	攻击者资源		目标状态	攻击者资源
P0	S0=S	V0=Ø	P36	S36={漏洞利用}	V36={H4（User）}
P11	S11=S	V11={基本信息}	P37	S37={漏洞利用}	V37={H4（Root）}
P12	S12=S	V12={社工信息}	P38	S38={漏洞利用}	V38={H3（User）}
P1	S1=S	V1={漏洞列表}	P39	S39={漏洞利用}	V39={H3（Root）}
P21	S21=S	V21={恶意代码}	P3	S3={漏洞利用}	V3={H6（Root）}
P22	S22=S	V22={服务器}	P41	S41=S3	V41={回传通道}
P2	S2=S	V2={‘钓鱼’}	P42	S42={漏洞利用}	V42={H5（User）}
P31	S31={漏洞利用}	V31={H2（User）}	P4	S4=S41-{信息}	V4={信息}
P32	S32={漏洞利用}	V32={H8（Root）}	P41	S41=S3	V41={回传通道}
P33	S33={漏洞利用}	V33={H7（User）}	P5	S5=S4-{M}	V5={M}
P34	S34={漏洞利用}	V34={H7（Root）}	P51	S51=S4	V51={H5（Root）}
P35	S35={漏洞利用}	V35={H1（Root）}	P6	S6=S-{M}	V6={M}

表 7　实验中变迁及变迁的前驱、后继库所列表

变迁含义	前驱库所	后继库所	变迁含义	前驱库所	后继库所
人肉搜索	P0	P11	CVE-2013-3940	P2 P35	P36
社交网络	P0	P12	CVE-2013-5065	P35 P36	P37
漏洞扫描	P11 P12	P1	CVE-2008-0407	P2	P38
制造恶意代码	P1	P21	CVE-2014-0407	P36 P37 P38	P39
伪造服务器	P1	P22	CVE-2010-0249	P2	P3
设置“钓鱼”	P21 P22	P2	CVE-2013-3346	P2 P3	P42
CVE-2008-4250	P2	P31	CVE-2014-4971	P2 P3 P42	P51
CVE-2012-1853	P2	P32	建立回传通道	P3	P41
CVE-2010-1117	P2 P32	P33	信息回传	P41	P4
CVE-2015-0084	P32 P33	P34	渗透进入	P4	P51
CVE-2013-2249	P2	P35	窃取信息	P51	P5
CVE-2012-2687	P33 P34 P31	P35	痕迹清除	P5	P6
CVE-2011-3607	P33 P34 P31	P35			

3.3　实验结果分析

根据局域网中主机与服务器的连通情况构建其连通序列，具体的连通序列见表 8。然后构建实验局域网中的变迁序列。实验中设各主机及服务器的初始库所状态为<S0,V0>，且 S0=S,V0=Ø，即初始情况下，各目标状态为 S，攻击者获得资源为 Ø。根据实验中得到的变迁的前驱和后继表，得到变迁序列表，如表 9 所示。

对于库所序列，由于库所序列是随着变迁的发生而不断更新及改变的，所以为直观地描述出其由攻击开始到攻击完成的状态变化，以各主机及服务器的权限变化来体现实验中的库所变化。初始时，各主机及服务器的权限均为外部用户，随着变迁的发生，库所的状态也会发生变化，设外部用户为 0，User 为 1，Root 为 2，则构建主机、服务器与变迁的权限变化矩阵，其中每一列表示在当前变迁下的各主机及服务器的权限，并且只标记发生变化的主机及服务器，对于在该变迁下未发生变化的记为 0。使用模型生成算法，根据上文中的推理过程，根据生成的连通序列、变迁序列以及库所变化矩阵，生成当攻击目标为邮件服务器时的 APT 攻击路径，如图 10 所示。

在上述模拟极光攻击生成的 APTPN 模型中，共含潜在的攻击路径 107 条。对比真实的极光攻击，其中模型中的目标扫描阶段与真实的极光攻击完全吻合，即模型中 P0→P1 的攻击过程；真实极光攻击的工具定制与投递阶段是通过伪造带有钓鱼网页的虚假服务器来向主机 2 发送恶意代码的，即模型中 P1→P22 的攻击过程；真实的极光攻击的潜伏提权阶段采用的攻击路径是由外网直接攻击主机 2，并经由主机 2 渗透进入邮件服务器，即模型中 P2→P3 的攻击过程；通信控制阶段是通过建立 SSL 隧道进行信息回传的，即模型中 P3→P4 的攻击过程；模型中的目标达成阶段与撤收维护阶段也与真实的极光攻击完全吻合，即模型中 P4→P51→P5→P6 的攻击过程。

表 8　连通序列表

连通序列		
<192.168.42.88,192.168.193.130,1>	<192.168.117.111,192.168.120.222,1>	<192.168.117.111,192.168.117.222,1>
<192.168.193.129,192.168.117.222,1>	<192.168.120.111,192.168.120.222,1>	<192.168.117.111,192.168.120.111,1>
<192.168.193.129,192.168.117.111,1>	<192.168.193.129,192.168.42.88,1>	<192.168.120.111,192.168.42.88,1>
<192.168.193.129,192.168.120.111,1>	<192.168.193.129,192.168.42.111,1>	<192.168.120.111,192.168.42.111,1>
<192.168.193.129,192.168.120.222,1>	<192.168.117.111,192.168.42.88,1>	
<192.168.193.129,192.168.193.130,1>	<192.168.117.111,192.168.42.111,1>	

表 9　变迁序列表

变迁序列					
<P0,P11,T111,2>	<P32,P33,T313,2>	<P35,P36,T315,2>	<P3,P42,T413,2>	<P22,P2,T221,2>	<P33,P35,T333,2>
<P0,P12,T112,2>	<P32,P34,T321,2>	<P35,P37,T351,2>	<P2,P5,T412,2>	<P2,P31,T311,2>	<P34,P35,T333,2>
<P11,P1,T121,2>	<P33,P34,T331,2>	<P36,P37,T351,2>	<P3,P41,T411,2>	<P2,P32,T312,2>	<P31,P35,T333,2>
<P12,P1,T121,2>	<P2,P35,T314,2>	<P2,P38,T316,2>	<P41,P4,T421,2>	<P2,P33,T313,2>	<P2,P36,T315,2>
<P1,P21,T211,2>	<P33,P34,T332,2>	<P36,P39,T361,2>	<P3,P5,T412,2>	<P2,P3,T317,2>	<P2,P3,T317,2>
<P1,P22,T212,2>	<P34,P35,T332,2>	<P37,P39,T361,2>	<P42,P5,T412,2>	<P38,P3,T362,2>	<P38,P3,T362,2>
<P21,P2,T221,2>	<P31,P35,T332,2>	<P38,P39,T361,2>	<P4,P51,T511,2>	<P39,P3,T362,2>	<P2,P42,T413,2>

$$
\begin{array}{cccccccccccccccccccccccc}
 & \cdots & T_{221} & T_{311} & T_{312} & T_{313} & T_{314} & T_{315} & T_{316} & T_{317} & T_{321} & T_{331} & T_{332} & T_{333} & T_{351} & T_{361} & T_{362} & T_{411} & T_{412} & T_{421} & T_{511} & T_{521} & T_{611} \\
H_1 & \cdots & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
H_2 & & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
H_3 & & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
H_4 & \cdots & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
H_5 & & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & 0 & 0 \\
H_6 & & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\
H_7 & & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
H_8 & \cdots & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}
$$

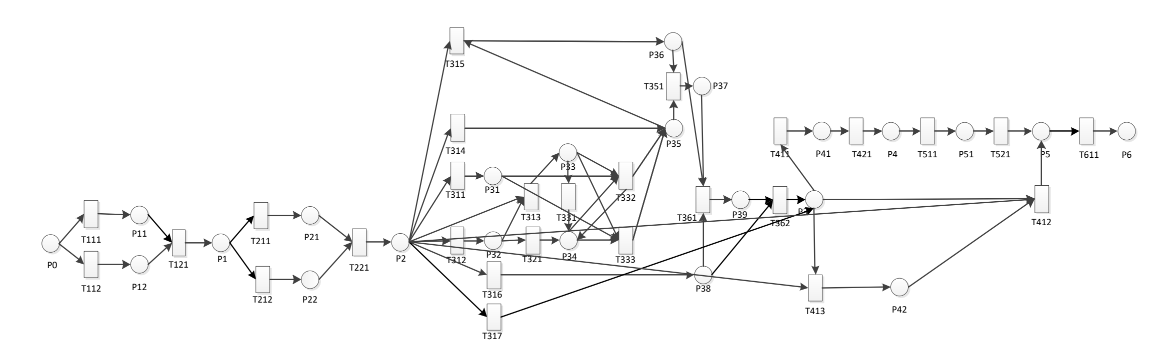


图 10　模拟极光攻击的 APTPN 模型

其余的攻击路径则是可能发生但却在真实的极光攻击中并未发生的攻击行为，但由于 APT 攻击的持续性和长期性，这些潜在的攻击路径能够为 APT 攻击的防御提供依据。通过在攻击发生之前，对其潜在攻击路径的防御，能够减小 APT 攻击的不确定性带来的危害。

同时，在实验用内主机侧和流量侧收集相关信息，得到如表 10 所示的信息集。

表 10　实验主机及流量侧收集信息

编号	时间	内容	源 IP	目的 IP	源端口	目的端口
1	19:32	IPsweep	192.168.42.11	192.168.117.111	443	1344
2	19:33	portsweep	192.168.42.11	192.168.117.111	443	1344
3	19:00	收到好友的邮件		192.168.42.11		
4	19:30	访问未知 DNS 域名	192.168.117.111	192.168.42.11	80	443
5	19.31	浏览器溢出	192.168.42.11	192.168.117.111	443	1157
6	19.32	CPU 消耗异常	192.168.42.11	192.168.117.111	443	1157
7	21:22	未知程序执行	192.168.42.11	192.168.117.111	443	1344
8	21:23	注册表添加未知键值				
9	21:25	未知探测数据包发送	192.168.117.111	192.168.42.11	1344	443

根据表 10 中信息，可根据信息 3 判定此时已完成库所是 P2，而 P2 的后置变迁在模型中包含 9 个为{ T311, T312, T313, T314, T315, T316, T317, T412, T413}，这些变迁均可能是攻击者下一步的攻击手段。通过进一步收集信息，可由信息 4、5 及 6 判定此时已完成的库所是 P3，而后置变迁只有三个为 T411、T412、T413。其含义为建立回传通道和相关漏洞的漏洞利用。由此可以根据基于 APTPN，以判定为已发生的库所为起点，预测攻击者下一步的攻击手段以及检测是否已发生 APT 攻击。

实验中，采用的是专家系统法对库所进行判定，该方法主观性较强，而在实际工作中可利用 2.3 节提到的方法计算库所发生的概率，减少主观性，增加准确率和提高说服力。

4　结束语

本文的工作包含三个部分：a)根据 APT 攻击特点建立基于 Petri 网的 APT 攻击模型；b)设计 APTPN 模型生成算法；c)模拟极光攻击，利用模型及模型生成算法，生成极光攻击的 APTPN 模型。创新点有两个：a)结合以往网络攻击的 Petri 网模型，设计并实现针对 APT 攻击的 Petri 网模型，通过对库所及变迁的映射，该模型能够较好地体现出 APT 攻击的针对性、阶段性、持续性特点；b)依据该模型，设计并实现 APTPN 模型生成算法，该算法能够自动生成针对特定局域网内攻击目标的 APT 攻击路径，并且算法复杂度可以控制在 O（N^2）。

但是本文算法的前提是建立在同一 APT 组织的攻击目标不变的情况下，而对于不同 APT 组织的不同攻击目标，算法并不适用。因此，下一步的工作应着重研究模型生成算法的扩展，同时针对生成的攻击路径进行量化，预测其发生的可能性。

参考文献：

[1] 付钰，李洪成，吴晓平，等. 基于大数据分析的 APT 攻击检测研究综述 [J]. 通信学报, 2015 (11): 1-14.

[2] 张瑜，潘小明, LIU Qingzhong，等. APT 攻击与防御 [J]. 清华大学学报: 自然科学版, 2017, 57 (11): 1127-1133.

[3] 李建方，张士铎. 高级可持续威胁攻击关键技术探究 [J]. 中国传媒大学学报: 自然科学版, 2016 (3): 51-55.

[4] Xiaomei Li. Research on prevention solution of advanced persistent threat [C]// Proc of the 2nd International Conference on Software Engineering, Knowledge Engineering and Information Engineering. Singapore: Springer, 2014: 1-4.

[5] Murata T. Petri nets: properties, analysis and applications [J]. Proceedings of the IEEE. 1989, 77 (4): 541-580.

[6] Zhao Wentao, Wang Pengfei. Zhang Fan. Extended petri net-based advanced persistent threat analysis model [C]// Proc of International Conference on Computer Engineering and Network. Heidelberg: Springer, 2014: 1297-1305.

[7] 罗森林，张驰，周梦婷，等. 基于时间 Petri 网的渗透测试攻击模型研究 [J]. 北京理工大学学报. 2015 (1): 92-96.

[8] 黄光球，白璐. 基于对象 Petri 网的信任攻击建模与分析 [J]. 系统仿真学报. 2017 (8): 1702-1711.

[9] 吴迪，连一峰，陈恺，等. 一种基于攻击图的安全威胁识别和分析方法 [J]. 计算机学报. 2012 (9): 1938-1950.

[10] 黄永洪，吴一凡，杨豪璞，等. 基于攻击图的 APT 脆弱节点评估方法 [J]. 重庆邮电大学学报: 自然科学版, 2017 (4): 535-541.

[11] 王辉，杨光灿，韩冬梅. 基于贝叶斯网络的内部威胁预测研究 [J]. 计

算机应用研究, 2013, 30 (9): 2767-2771.

[12] Hutchins E M, Cloppert M J, Amin R M. LM-white-paper-intel-driven-defense [R//OL]. (2011) .

[13] Peterson J. Petri 网理论与系统模拟 [M]. 吴哲辉, 译. 徐州: 中国矿业大学出版社, 1989.

[14] 周建涛, 叶新铭. Petri 网的可达图与可达树的比较 [J]. 内蒙古大学学报: 自然科学版, 2000, 31 (1): 117-120.

[15] 张艳雪, 赵冬梅, 刘金星. 基于模糊—隐马尔可夫模型的复合式攻击预测方法 [J]. 电光与控制, 2015 (1): 39-44.

[16] 杜镇宇, 李翼宏, 张亮. APT 样本逻辑表达式生成算法 [J]. 计算机工程与应用, 2018 (1): 1-10.